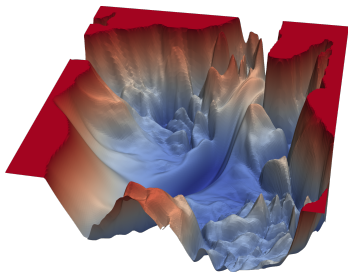


# Introduction to Machine Learning

## Properties of Loss Functions



### Learning goals

- Know the concept of robustness
- Learn about analytical and computational properties of loss functions
- Understand that the loss function may influence convergence of the optimizer

# THE ROLE OF LOSS FUNCTIONS

Why should we care about how to choose the loss function  $L(y, f(\mathbf{x}))$ ?

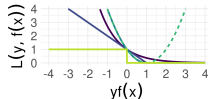
- **Statistical** properties: choice of loss implies statistical assumptions on the distribution of  $y \mid \mathbf{x} = \mathbf{x}$  (see *maximum likelihood estimation vs. empirical risk minimization*).
- **Robustness** properties: some loss functions are more robust towards outliers than others.
- **Analytical** properties: the computational / optimization complexity of the problem

$$\arg \min_{\theta \in \Theta} \mathcal{R}_{\text{emp}}(\theta)$$

is influenced by the choice of the loss function.

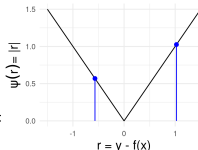
# BASIC TYPES OF REGRESSION LOSSES

- Regression losses usually only depend on the **residuals**  $r := y - f(\mathbf{x})$ .
- Classification losses are usually expressed in terms of the **margin**  $\nu := y \cdot f(\mathbf{x})$ .
- Losses are called **symmetric** if  $L(y, f(\mathbf{x})) = L(f(\mathbf{x}), y)$ .
- A loss is **translation-invariant** if  $L(y + a, f(\mathbf{x}) + a) = L(y, f(\mathbf{x}))$ ,  $a \in \mathbb{R}$ .
- A loss is called **distance-based** if
  - it can be written in terms of the residual, i.e.,  $L(y, f(\mathbf{x})) = \psi(r)$  for some  $\psi : \mathbb{R} \rightarrow \mathbb{R}$ , and
  - $\psi(r) = 0 \Leftrightarrow r = 0$ .

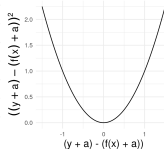


Exponential      Squared (scc)  
Hinge              0-1  
Squared hinge

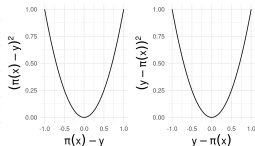
Margin-based losses



Distance-based: L1



Translation-invariant: L2



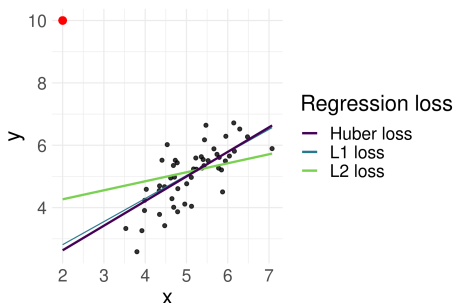
Symmetric: Brier score

# ROBUSTNESS

Outliers (in  $y$ ) have large residuals  $r = y - f(\mathbf{x})$ . Some losses are more strongly affected by large residuals than others.

| $y - \hat{f}(\mathbf{x})$ | $L1$ | $L2$ | Huber ( $\epsilon = 5$ ) |
|---------------------------|------|------|--------------------------|
| 1                         | 1    | 1    | 0.5                      |
| 5                         | 5    | 25   | 12.5                     |
| 10                        | 10   | 100  | 37.5                     |
| 50                        | 50   | 2500 | 237.5                    |

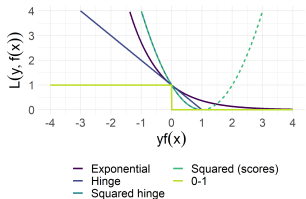
As a consequence, a model is less influenced by outliers than by inliers if the loss is **robust**.



$L2$  is an example for a loss function that is not very robust towards outliers. It penalizes large residuals more than  $L1$  or Huber loss, which are considered robust.

# ANALYTICAL PROPERTIES: SMOOTHNESS

- **Smoothness** of a function is a property measured by the number of continuous derivatives.
- A function is said to be  $\mathcal{C}^k$  if it is  $k$  times continuously differentiable. A function is  $\mathcal{C}^\infty$  if it is continuously differentiable for all orders  $k$ .
- Derivative-based methods require a certain level of smoothness of the risk function  $\mathcal{R}_{\text{emp}}(\theta)$ .
  - If the loss function is not smooth, the risk minimization problem will generally not be smooth either.
  - This may require the use of derivative-free optimization (which might not be desirable).
  - Smoothness of objective wrt  $\theta$  not only depends on  $L(y, f(\mathbf{x}))$ , but also requires smoothness of  $f(\mathbf{x})$ !



Squared loss, exponential loss and squared hinge loss are continuously differentiable.  
Hinge loss is continuous but not differentiable.  
0-1 loss is not even continuous.

# ANALYTICAL PROPERTIES: SMOOTHNESS

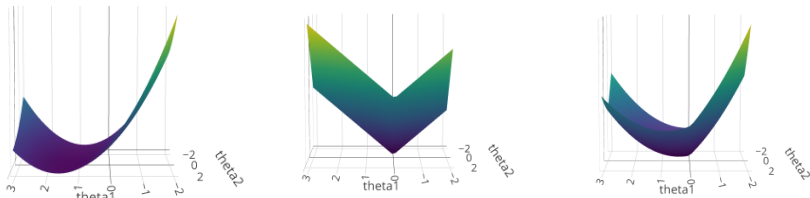
## Example: Lasso regression

- Problem: Lasso has a non-differentiable objective function

$$\mathcal{R}_{\text{reg}}(\boldsymbol{\theta}) = \|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|_2^2 + \lambda\|\boldsymbol{\theta}\|_1 \in \mathcal{C}^0,$$

but many optimization methods are derivative-based, e.g.,

- Gradient descent: requires existence of gradient  $\nabla \mathcal{R}_{\text{emp}}(\boldsymbol{\theta})$ ,
- Newton-Raphson: requires existence of Hessian  $\nabla^2 \mathcal{R}_{\text{emp}}(\boldsymbol{\theta})$ .
- We must therefore resort to alternative optimization techniques – for instance, coordinate descent with subgradients.



Example:  $y = x_1 + 1.2x_2 + \epsilon$ . *Left*: unpenalized objective, *middle*: L1 penalty, *right*: penalized objective (all as functions of  $\boldsymbol{\theta}$ ). We see how the L1 penalty nudges the optimum towards (0, 0) and compromises the original objective's smoothness.

# ANALYTICAL PROPERTIES: CONVEXITY

- A function  $\mathcal{R}_{\text{emp}}(\boldsymbol{\theta})$  is convex if

$$\mathcal{R}_{\text{emp}}\left(t \cdot \boldsymbol{\theta} + (1 - t) \cdot \tilde{\boldsymbol{\theta}}\right) \leq t \cdot \mathcal{R}_{\text{emp}}(\boldsymbol{\theta}) + (1 - t) \cdot \mathcal{R}_{\text{emp}}(\tilde{\boldsymbol{\theta}})$$

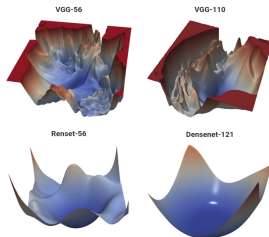
$\forall t \in [0, 1], \boldsymbol{\theta}, \tilde{\boldsymbol{\theta}} \in \Theta$  (strictly convex if the above holds with strict inequality).

- In optimization, convex problems are desirable because they have a number of convenient properties. In particular, all local minima are global.  
→ strictly convex function has at most **one** global minimum (uniqueness).
- For  $\mathcal{R}_{\text{emp}} \in \mathcal{C}^2$ ,  $\mathcal{R}_{\text{emp}}$  is convex iff Hessian  $\nabla^2 \mathcal{R}_{\text{emp}}(\boldsymbol{\theta})$  is psd.

# ANALYTICAL PROPERTIES: CONVEXITY

- Convexity of  $\mathcal{R}_{\text{emp}}(\theta)$  depends both on convexity of  $L(\cdot)$  (given in most cases) and  $f(\mathbf{x} \mid \theta)$  (often problematic).
- If we model our data using an exponential family distribution, we always get convex losses
  - For  $f(\mathbf{x} \mid \theta)$  linear in  $\theta$ , linear/logistic/softmax/poisson regression are convex problems!

Li et al., 2018: *Visualizing the Loss Landscape of Neural Nets*. The problem on the bottom right is convex, the others are not (note that very high-dimensional surfaces are coerced into 3D here).





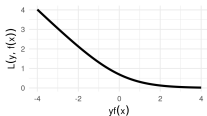
# ANALYTICAL PROPERTIES: CONVERGENCE

The choice of the loss function may also impact convergence behavior.

In the extreme case of **complete separation**, optimization might even fail entirely. Consider the following scenario:

- Margin-based loss that is antitonic in  $y \cdot f$  – for example, **Bernoulli loss**:

$$L(y, f(\mathbf{x})) = \log(1 + \exp(-yf(\mathbf{x})))$$



- Data perfectly separable by our learner

$$\Rightarrow y^{(i)} f(\mathbf{x}^{(i)} | \theta) > 0 \quad \forall \mathbf{x}^{(i)} \neq \mathbf{0}$$

as every  $\mathbf{x}^{(i)}$  is correctly classified:  $f(\mathbf{x}^{(i)} | \theta) < 0$  for  $y^{(i)} = -1$ ,  $> 0$  for  $y^{(i)} = 1$

$$\Rightarrow yf(\mathbf{x} | \theta) = |f(\mathbf{x} | \theta)|$$

- $f$  linear in  $\theta$  – for example, **logistic regression** with  $f(\mathbf{x} | \theta) = s(\theta^\top \mathbf{x})$

# ANALYTICAL PROPERTIES: CONVERGENCE

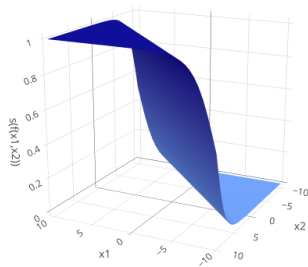
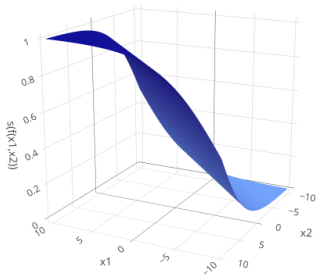
- In optimization, e.g., with gradient descent, we can always find a set of parameters  $\theta'$  that classifies all samples perfectly.
- But taking a closer look at  $\mathcal{R}_{\text{emp}}(\theta)$ , we find that the same can be achieved with  $2 \cdot \theta'$  – and at lower risk:

$$\begin{aligned}\mathcal{R}_{\text{emp}}(2 \cdot \theta) &= \sum_{i=1}^n L\left(\left|f\left(\mathbf{x}^{(i)} \mid 2 \cdot \theta\right)\right|\right) = \sum_{i=1}^n L\left(2 \cdot \left|f\left(\mathbf{x}^{(i)} \mid \theta\right)\right|\right) \\ &< \sum_{i=1}^n L\left(\left|f\left(\mathbf{x}^{(i)} \mid \theta\right)\right|\right) = \mathcal{R}_{\text{emp}}(\theta)\end{aligned}$$

- This actually holds true for every  $a \cdot \theta$  with  $a > 1$ .  
 $\Rightarrow$  By increasing  $\|\theta\|$ , our loss becomes smaller and smaller, and optimization runs infinitely.

# ANALYTICAL PROPERTIES: CONVERGENCE

- Geometrically, this translates to an ever steeper slope of the logistic/softmax function, i.e., increasingly sharp discrimination:



- In practice, data are seldom linearly separable and misclassified examples act as counterweights to increasing parameter values.
- Besides, we can apply **regularization** to encourage convergence to robust solutions.